

000474

# INCOMPRESSIBLE NAVIER-STOKES SOLVERS IN PRIMITIVE VARIABLES AND THEIR APPLICATIONS TO STEADY AND UNSTEADY FLOW SIMULATIONS

Cetin C. Kiris, [kiris@nas.nasa.gov](mailto:kiris@nas.nasa.gov)  
Dochan Kwak, [dkwak@mail.arc.nasa.gov](mailto:dkwak@mail.arc.nasa.gov)  
Stuart E. Rogers, [rogers@nas.nasa.gov](mailto:rogers@nas.nasa.gov)

NAS Applications Branch, M.S, T27B-1  
NASA-Ames Research Center, Moffett Field, CA 94035

This paper reviews recent progress made in incompressible Navier-Stokes simulation procedures and their application to problems of engineering interest. Discussions are focused on the methods designed for complex geometry applications in three dimensions, and thus are limited to primitive variable formulation. A summary of efforts in flow solver development is given followed by numerical studies of a few example problems of current interest. Both steady and unsteady solution algorithms and their salient features are discussed. Solvers discussed here are based on a structured-grid approach using either a finite-difference or a finite-volume frame work. As a grand-challenge application of these solvers, an unsteady turbopump flow simulation procedure has been developed which utilizes high performance computing platforms. In the paper, the progress toward the complete simulation capability of the turbo-pump for a liquid rocket engine is reported. The Space Shuttle Main Engine (SSME) turbo-pump is used as a test case for evaluation of two parallel computing algorithms that have been implemented in the INS3D code. The relative motion of the grid systems for the rotor-stator interaction was obtained using overset grid techniques. Unsteady computations for the SSME turbo-pump, which contains 114 zones with 34.5 million grid points, are carried out on SGI Origin 3000 systems at NASA Ames Research Center. The same procedure has been extended to the development of NASA-DeBakey Ventricular Assist Device (VAD) that is based on an axial blood pump. Computational, and clinical analysis of this device are presented.

## INTRODUCTION

Incompressible flow can be considered as a limiting case of compressible flow as the flow speed approaches to a significantly low value compared to the speed of sound. There are a large number of flow problems of practical importance in aerospace and other fields which belong in this category. The incompressible Navier-Stokes equations, which govern these flows, pose a special problem of satisfying the mass conservation equation because it is not coupled to the momentum equations. Physically, these equations are characterized by the elliptic behavior of the pressure waves, the speed of which are infinite.

Various methods have been developed, which can be classified in numerous ways depending on the choice of formulations, variables, or algorithms. Since three-dimensional applications involving complex geometries are of our primary interest, the primitive variable formulation is chosen in the present study. The primitive variables, namely the pressure and the velocities, can easily be defined in real geometry compared to derived quantities like stream function or vorticity. Therefore, for convenience and flexibility, primitive variable formulations were used for developing incompressible Navier-Stokes codes (INS3D family of codes) at NASA Ames Research Center. The present article is intended to present our progress made since the review given by the second author in 1989 as a VKI Lecture note (Kwak, 1989). The solution procedures presented here are mainly within a structured-grid framework. During the last

several years, a large number of review articles and books on CFD discussed incompressible flow methods. For a more comprehensive review of computational methods for incompressible flow in general, readers are referred to these materials, i.e. Hirsch (1988), Hafez and Oshima (1995).

## **SOLUTION METHODS**

In this section, two solution methods used in the development of INS3D, namely, an artificial compressibility method and a pressure projection method, are reviewed. The governing equations will be given first, followed by a discussion on the computational procedure related to the two methods. Three-dimensional incompressible flow with constant density is governed by the following Navier-Stokes equations:

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (2.1)$$

$$\frac{\partial u_i}{\partial t} + \frac{\partial u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} \quad (2.2)$$

where  $t$  is the time,  $x_i$  the Cartesian coordinates,  $u_i$  the corresponding velocity components,  $P$  the pressure, and  $\tau_{ij}$  the viscous-stress tensor. All the variables have been non-dimensionalized by a reference velocity and length scale. The viscous stress tensor can be written as

$$\tau_{ij} = 2\nu S_{ij} - R_{ij} \quad (2.3)$$

$$S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (2.4)$$

where,  $\nu$  is the kinematic viscosity,  $S_{ij}$  is the strain-rate tensor, and  $R_{ij}$  are the Reynolds stresses. Various levels of closure models for  $R_{ij}$  are possible. In the present article, turbulence is simulated by an eddy viscosity model using a constitutive equation of the following form:

$$R_{ij} = \frac{1}{3} R_{kk} \delta_{ij} - 2\nu_t S_{ij} \quad (2.5)$$

where  $\nu_t$  is the turbulent eddy viscosity. By including the normal stress,  $R_{kk}$ , in the pressure,  $\nu$  in equation (2.3) can be replaced by  $(\nu + \nu_t)$  as follows.

$$\tau_{ij} = 2(\nu + \nu_t) S_{ij} = 2\nu_t S_{ij} \quad (2.6)$$

In the remainder of this note the total viscosity,  $(\nu + \nu_t)$ , will be represented simply by  $\nu$ . The present formulations allow for spatially varying viscosity.

In general curvilinear coordinates,  $(\xi, \eta, \zeta)$ , the governing equations can be written as

$$\frac{\partial}{\partial t} \hat{u} = -\frac{\partial}{\partial \xi_i} (\hat{e}_i - \hat{e}_n) + \hat{s} = -\hat{r} \quad (2.7)$$

$$\frac{\partial}{\partial \xi_i} \left( \frac{U - (\xi)_i}{J} \right) = 0 \quad (2.8)$$

where  $\hat{r}$  is the right-hand side of the momentum equation, and

$$\xi_i = \xi, \eta \text{ or } \zeta, \text{ for } i=1, 2, \text{ or } 3$$

$$\begin{aligned}
\hat{u} &= \frac{1}{J} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \\
\hat{e}_i &= \frac{1}{J} \begin{bmatrix} (\xi_i)_x p + u U_i \\ (\xi_i)_y p + v U_i \\ (\xi_i)_z p + w U_i \end{bmatrix}, \quad \hat{e}_v = \frac{v}{J} \nabla \xi_i \cdot \left( \nabla \xi_i \frac{\partial}{\partial \xi_i} \right) \begin{bmatrix} u \\ v \\ w \end{bmatrix} \\
U_i &= (\xi_i)_x u + (\xi_i)_y v + (\xi_i)_z w \\
J &= \text{Jacobian of the transformation} \\
\hat{s} &= \text{source term}
\end{aligned} \tag{2.9}$$

The source term  $\hat{s}$  is used to represent centrifugal and Coriolis forces in a steady rotating reference frame, and will be discussed later in this section. For most flow applications, this term is set to zero.

### ARTIFICIAL COMPRESSIBILITY METHOD

Major advances in the state of the art in CFD have been made in conjunction with compressible flow computations. Therefore, it is of significant interest to be able to use some of these compressible flow algorithms for incompressible flows. To do this, the artificial compressibility method of Chorin (1967) can be used. In this formulation, the continuity equation is modified by adding a pseudo-time derivative of the pressure, resulting in

$$\frac{1}{\beta} \frac{\partial p}{\partial \tau} + \frac{\partial u_i}{\partial x_i} = 0 \tag{2.10}$$

where  $\beta$  is an artificial compressibility parameter and  $\tau$  is a pseudo-time parameter. This forms a hyperbolic-parabolic type of pseudo-time dependent system of equations. Thus, implicit schemes developed for compressible flows can be implemented to solve for steady-state solution. In the steady-state formulation the equations are to be marched in a time-like fashion until the divergence of velocity in equation (2.10) converges to a specified tolerance. The time variable for this process no longer represents physical time, so in the momentum equations  $t$  is replaced with  $\tau$ , which can be thought of as a pseudo-time or iteration parameter.

Physically, this means that waves of finite speed are introduced into the incompressible flow field as a medium to distribute the pressure. For a truly incompressible flow, the wave speed is infinite, whereas the speed of propagation of these pseudo waves depend on the magnitude of the artificial compressibility parameter. In a truly incompressible flow, the pressure field is affected instantaneously by a disturbance in the flow, but with artificial compressibility, there is a time lag between the flow disturbance and its effect on the pressure field. Ideally, the value of the artificial compressibility parameter is to be chosen as high as the particular choice of algorithm will allow so that the incompressibility is recovered quickly. This has to be done without lessening the accuracy and the stability property of the numerical method implemented. On the other hand, if the artificial compressibility parameter is chosen such that these waves travel too slowly, then the variation of the pressure field accompanying these waves is very slow. This will interfere with the proper development of the viscous boundary layer. In viscous flows, the behavior of the boundary layer is very sensitive to the streamwise pressure gradient, especially when the boundary layer is separated. If separation is present, a pressure wave traveling with finite speed will cause a change in the local pressure gradient which will affect the location of the flow separation. This change in separated flow will feed back to the pressure field, possibly preventing convergence to a steady state. When the viscous effect is important for the entire flow field as in most internal flow problems, the interaction between the pseudo-pressure waves and the viscous flow field is especially important.

Artificial compressibility relaxes the strict requirement of satisfying mass conservation in each step. However, to utilize this convenient feature, it is essential to understand the nature of the artificial compressibility both physically and mathematically. Chang and Kwak (1984) reported details of the artificial compressibility, and suggested some guidelines for choosing the artificial compressibility parameter. Various applications which evolved from this concept

have been reported for obtaining steady-state solutions (e.g., Steger and Kutler, 1977; Kwak et al. 1986; Chang et al., 1988; Choi and Merkle, 1985). To obtain time-dependent solutions using this method, an iterative procedure can be applied in each physical time step such that the continuity equation is satisfied (see, Merkle and Athavale, 1987, Rogers and Kwak, 1988, Rogers, Kwak, and Kiris, 1991, Belov et. al., 1995). Further discussions on the artificial compressibility approach can be found in the literature (see, Temam, 1979, Rizzi and Eriksson, 1985).

Combining equation (2.10) and the momentum equations gives the following system of equations:

$$\frac{\partial}{\partial \tau} \hat{D} = - \frac{\partial}{\partial \xi_i} (\hat{E}_i - \hat{E}_{vi}) + \hat{S} = -\hat{R} \quad (2.11)$$

where  $\hat{R}$  is the right-hand-side of the momentum equation and can be defined as the residual for steady-state computations, and where

$$\hat{D} = \frac{D}{J} = \frac{1}{J} \begin{bmatrix} p \\ u \\ v \\ w \end{bmatrix}, \quad \hat{E} = \begin{bmatrix} \beta U_i / J \\ \hat{e}_i \end{bmatrix}, \quad \hat{E}_{vi} = \begin{bmatrix} 0 \\ \hat{e}_{vi} \end{bmatrix} \quad (2.12)$$

When the governing equations are solved in a steadily rotating reference frames, the source term,  $\hat{S}$ , represents centrifugal and Coriolis terms. If the relative reference frame is rotating around the x-axis, the source term  $\hat{S}$  is given by

$$\hat{S} = \begin{bmatrix} 0 \\ 0 \\ \Omega(\Omega y + 2w) \\ \Omega(\Omega z - 2v) \end{bmatrix}$$

where  $\Omega$  is the rotational speed. In this report, the source term,  $\hat{S}$ , is set to zero other than for rotational steady solutions. Relative velocity components are written in terms of absolute velocity components  $u_a$ ,  $v_a$ , and  $w_a$  as

$$\begin{aligned} u &= u_a \\ v &= v_a + \Omega z \\ w &= w_a - \Omega y \end{aligned}$$

Time-dependent calculation of incompressible flows are especially time consuming due to the elliptic nature of the governing equations. This means that any local change in the flow has to be propagated throughout the entire flow field. Numerically, this means that in each time step, the pressure field has to go through one complete steady-state iteration cycle, for example, by Poisson-solver-type pressure iteration or artificial compressibility iteration method. In transient flow, the physical time step has to be small and consequently the change in the flow field may be small. In this situation, the number of iterations in each time step for getting a divergence-free flow field may not be as high as regular steady-state computations. However, the time-accurate computations are generally an order of magnitude more time-consuming than steady-state computation. Therefore, it is particularly desirable to develop computationally efficient methods either by implementing a fast algorithm and by utilizing computer characteristics such as vectorization and parallel processing.

A time-accurate method using artificial compressibility developed by Rogers, Kwak, and Kiris (1991) is summarized next. In this formulation the time derivatives in the momentum equations are differenced using a second-order, three-point, backward-difference formula

$$\frac{3\hat{u}^{n+1} - 4\hat{u}^n + \hat{u}^{n-1}}{2\Delta t} = -\hat{r}^{n+1} \quad (2.13)$$

where the superscript  $n$  denotes the quantities at time  $t = n\Delta t$  and  $\hat{r}$  is the right-hand side given in equation (2.7). To solve equation (2.13) for a divergence free velocity field at the  $(n+1)$  time level, a pseudo-time level is introduced and is denoted by a superscript  $m$ . The equations are iteratively solved such that  $\hat{u}^{n+1,m+1}$  approaches the new velocity  $\hat{u}^{n+1}$  as the divergence of  $\hat{u}^{n+1,m+1}$  approaches zero. To drive the divergence of this velocity to zero, the following artificial compressibility relation is introduced:

$$\frac{p^{n+1,m+1} - p^{n+1,m}}{\Delta \tau} = -\beta \nabla \cdot \hat{u}^{n+1,m+1} \quad (2.14)$$

where  $\tau$  denotes pseudo-time and  $\beta$  is an artificial compressibility parameter. Combining equation (2.14) with the momentum equations gives

$$I_{ir} (\hat{D}^{n+1,m+1} - \hat{D}^{n+1,m}) = -\hat{R}^{n+1,m+1} - \frac{I_m}{\Delta t} (1.5\hat{D}^{n+1,m} - 2\hat{D}^n + 0.5\hat{D}^{n-1}) \quad (2.15)$$

where  $\hat{D}$  is the same vector defined in equation (2.13),  $\hat{R}$  is the same residual vector defined in equation (2.11), and  $I_{ir}$  is a diagonal matrix given by

$$I_{ir} = \text{diag} \left[ \frac{1}{\Delta t}, \frac{1.5}{\Delta t}, \frac{1.5}{\Delta t}, \frac{1.5}{\Delta t} \right]$$

Finally, the residual term at the  $m+1$  pseudo-time level is linearized giving the following equation in delta form

$$\left[ \frac{I_{ir}}{J} + \left( \frac{\partial \hat{R}}{\partial D} \right)^{n+1,m} \right] (\hat{D}^{n+1,m+1} - \hat{D}^{n+1,m}) = -\hat{R}^{n+1,m} - \frac{I_m}{\Delta t} (1.5\hat{D}^{n+1,m} - 2\hat{D}^n + 0.5\hat{D}^{n-1}) \quad (2.16)$$

As can be seen, this equation is very similar to the steady-state formulation which can be rewritten for the Euler implicit case as

$$\left[ \frac{1}{J\Delta \tau} I + \left( \frac{\partial \hat{R}}{\partial D} \right)^n \right] (\hat{D}^{n+1} - \hat{D}^n) = -\hat{R}^n \quad (2.17)$$

Both systems of equations will require the discretization of the same residual vector  $\hat{R}$ . The matrix equation is solved iteratively by using a non-factored Gauss-Seidel type line-relaxation scheme employed by MacCormack (1985), which maintains stability and allows a large pseudo-time step to be taken. Details of the numerical method can be found in paper by Rogers, Kwak, and Kiris (1991). The GMRES scheme has also been utilized for the solution of the resulting matrix equation (Rogers, 1995). Computer memory requirement for the flow solver (INS3D-UP code) with line-relaxation is 35 times the number of grid points in words, and with GMRES-ILU(0) scheme is 220 times the number of grid points in words.

## PRESSURE PROJECTION METHOD

In 1965, Harlow and Welch published the first primitive variable method using a Poisson equation for pressure. In this method, called the marker-and-cell (MAC) method, the pressure is used as a mapping parameter to satisfy the continuity equation. By taking the divergence of the momentum equation, the Poisson equation for pressure is obtained:

$$\nabla^2 p = \frac{\partial h}{\partial x_i} - \frac{\partial}{\partial t} \frac{\partial u_i}{\partial x_i} \quad (2.18)$$

where

$$h_i = -\frac{\partial u_i u_j}{\partial x_j} + \frac{\partial \tau_{ij}}{\partial x_j}$$

The usual computational procedure involves choosing the pressure field at the current time step such that continuity is satisfied at the next time step. The original MAC method is based on a staggered arrangement on a 2D Cartesian grid. The staggered grid conserves mass, momentum, and kinetic energy in a natural way and avoids odd-even point decoupling of the pressure encountered in a regular grid (Gresho and Sani, 1987). Even though the original method used an explicit Euler solver, various time advancing schemes can be implemented with this formulation. Ever since its introduction, numerous variations of the MAC method have been devised and successful computations have been made.

The MAC method can be viewed as a special case of the projection method (i.e. Chorin, 1968). In this method the strict requirement of obtaining the correct pressure for a divergence-free velocity field in each step may significantly slow down the overall computational efficiency. To satisfy the mass conservation in grid space, the difference form of the second derivative in the Poisson equation has to be constructed consistent with the discretized momentum equation (see Kwak, 1989).

To solve for a steady-state solution the correct pressure field is desired only when the solution is converged. In this case, the iteration procedure for the pressure can be simplified such that it requires only a few iteration at each time step. The best known method using this approach is the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) (Patankar, 1980 ; Chen et al., 1995). The unique feature of this method is the simple way of estimating the velocity and the pressure correction. This feature simplifies the computation but introduces empiricism into the method. Despite its empiricism, the method has been used successfully for many steady-state computations. It is not the intention of the present paper to evaluate this method, and readers interested in this approach are referred to the above cited references.

The time-integration scheme is based on operator splitting, which can be accomplished in several ways by combining the pressure, convective, and viscous terms in the momentum equations. The fractional-step method is based on the decomposition of vector field into a divergence free component and a gradient of a scalar field. Since its inception, this approach is perhaps the most widely used method in computing incompressible flow. Variations of this idea are too numerous to list here.

The common application of fractional-step method is done in two steps. The first step is to solve for an auxiliary velocity field using the momentum equations. In the second step, the velocity field is corrected by using the pressure, which can map the auxiliary velocity onto a divergence free velocity field. In the first step, the momentum equations are discretized in time using a second-order implicit Runge-Kutta method. The Poisson equation for pressure is obtained by taking the divergence of the momentum equations and by using the continuity equation. For the spatial discretization, a finite-volume formulation is used where pressure is defined at the cell center and the mass fluxes at the faces of each cell. The mass-conservation equation is evaluated by computing the mass flux across faces of a computational cell. When the mass fluxes are chosen as unknowns, the continuity equation is satisfied automatically in generalized coordinate systems. The continuity equation with this choice of the dependent variables takes a form identical to the Cartesian case. Therefore, the mass fluxes are considered as the *natural* dependent variables for projection methods in curvilinear coordinates. Treating the mass fluxes as dependent variables in a finite-volume formulation is equivalent to using contravariant velocity components, scaled by the inverse of the transformation Jacobian, in a finite-difference formulation. This choice of mass fluxes as dependent variables complicates the discretization of the momentum equations. In order to replace Cartesian velocity components by the new dependent variables, namely, the contravariant velocity components, the corresponding area vectors are dotted with the momentum equations. Then the integral momentum equation is evaluated on different computational cells for each unknown. For the definition of variables, a staggered grid orientation was selected to eliminate checker-board-like oscillations in pressure and provides more compact stencils. Full details on the derivation of momentum equations and the solution procedure is outlined in references by Rosenfeld, Kwak, and Vinukur (1991) and by Kiris and Kwak (1996). A flow solver using the above procedure is designated as INS3D-FS. Since each equation is solved in a segregated fashion, the memory requirement for GMRES solver in INS3D-FS is only 70 times the number of grid points in words.

## COMPUTED RESULTS

### FLOW PAST 90° FLAT PLATE

Numerical results for the time evolution of twin vortices behind a two-dimensional flat plate are presented. Several cases were run to with various algorithmic parameters. To expedite the process, a two-dimensional test case is selected here. It should be noted that the associated flow solvers, INS3D-UP for the artificial compressibility method and INS3D-FS for the pressure projection method, are written for three-dimensional applications. This numerical experiment is studied to help select a method for large three-dimensional unsteady applications where computing resources become a critical issue.

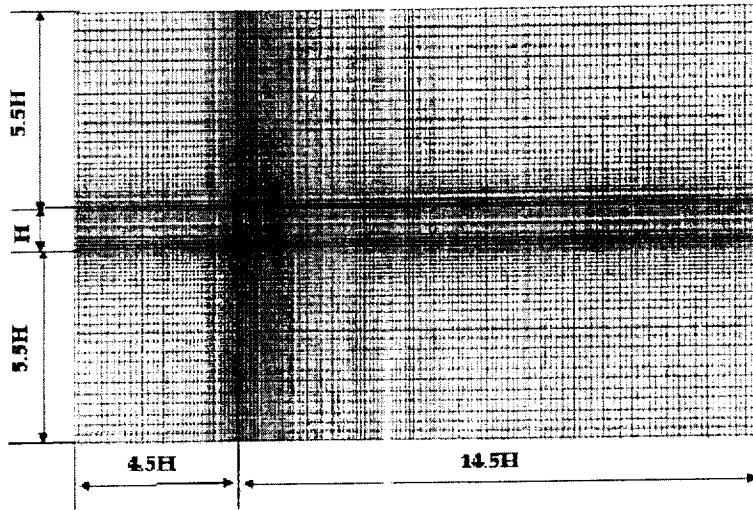


Figure 1. Computational grid for the flow past a 90-degree flat plate (plate thickness = 0.03 H).

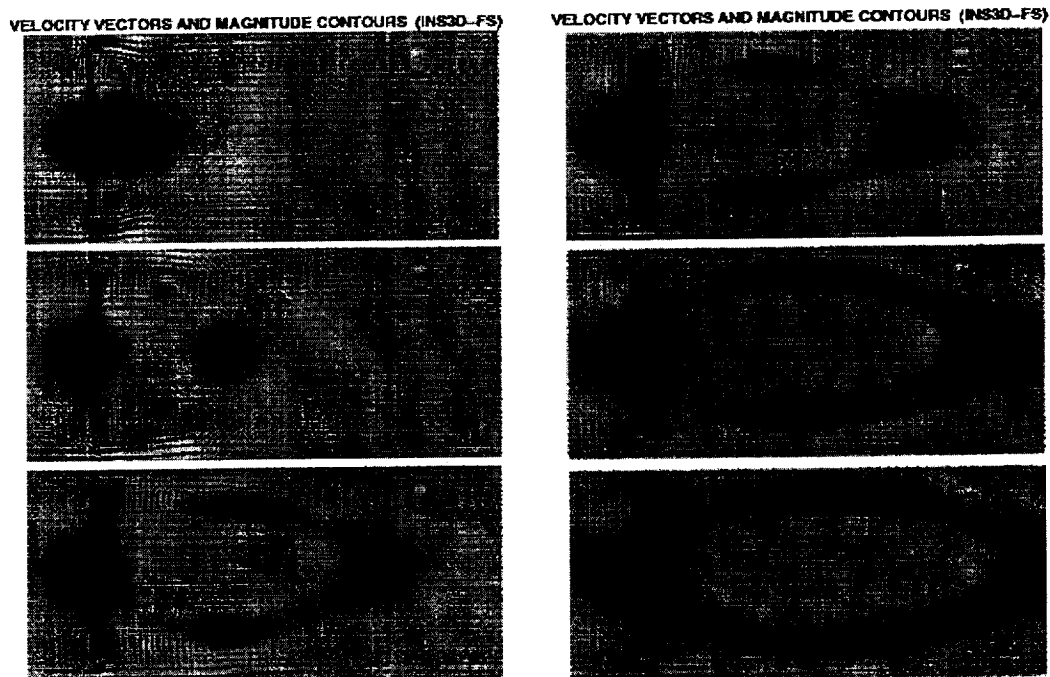


Figure 2. Velocity vectors at various non-dimensional times (INS3D-FS).

Computed results from both methods are compared with the experimental data by Taneda and Honji (1971). The experiment was carried out in a water tank 40 cm wide. A thin, 3 cm high plate was immersed in water. The flow was started from rest impulsively at the velocity  $u=0.495$  cm/s. The Reynolds number for this case is 126 based on the plate height. The computational grid size is  $181 \times 81$  in flow and vertical directions, and 3 layers of this grid are used to obtain two-dimensional results (figure 1). Since INS3D-FS is written in a finite-volume staggered-grid formulation, it requires one additional ghost cell in each direction. Figure 2 shows calculated velocity vectors obtained from INS3D-FS at various times. The flow separates at the edges of the plate and forms a vortex pair. The twin vortices become longer in the flow direction with time. The time history of the stagnation point is compared with experimental results and other numerical results in figure 3. Symbols represent experimental measurements, and the solid line and the dashed line represent results from INS3D-UP and INS3D-FS, respectively. In addition the dotted line shows the numerical results from finite element formulations of Yoshida and Nomura (1985). In figure 3, the interval for time integration was 0.5 sec, which corresponds to nondimensional value of 0.0825. Even though the plate started impulsively in the experiment, the computations presented in figure 3 have a slow start procedure. In figure 4, two different ways of starting the flow are prescribed, namely, an impulsive start as in figure 4a and a slow start as prescribed in figure 4b. Yoshida and Nomura (1985) used the same slow start procedure in their calculations. For the slow start case, the velocity profile shown in figure 4b is prescribed and the starting time of calculation is appropriately shifted from that of experiment.

First, INS3D-FS results are presented in figures 5 and 6. In figure 5, the effect of starting procedure on the development of the flow is shown. Here, a non-dimensional time step of 0.0825 was used. There are measurable differences in the resulting flows. Increasing the spatial resolution does not change the results significantly while decreasing the time step improves the agreement with experiments, as shown in figure 6.

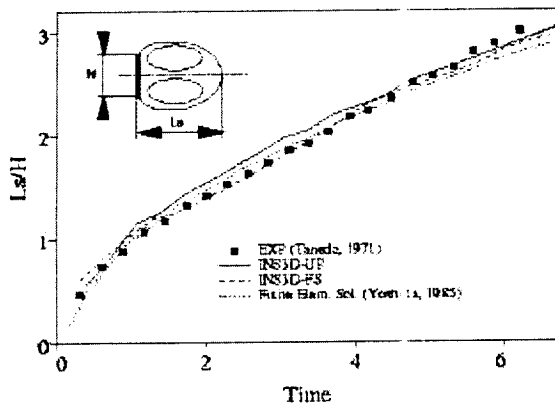


Figure 3. Calculated time history of the stagnation point.

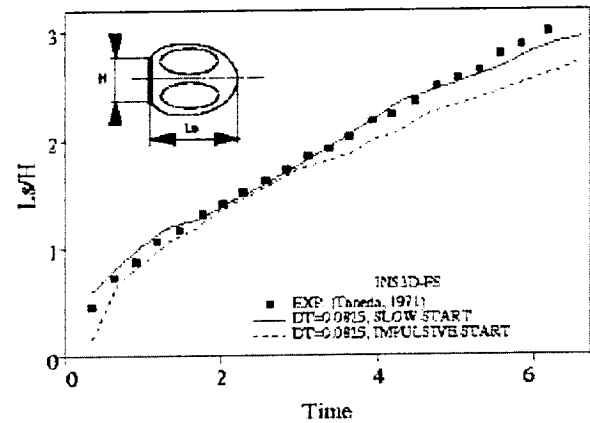


Figure 5. Effects of starting procedure.

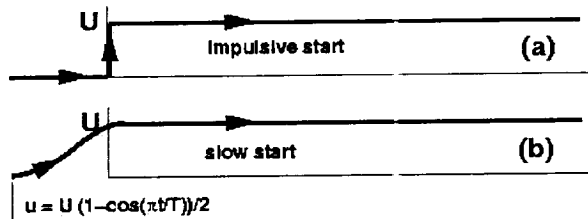
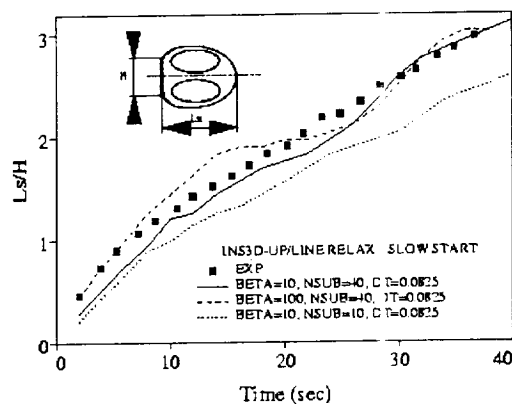


Figure 4. Prescribed velocity for an impulsive start (a) and for a slow start (b).

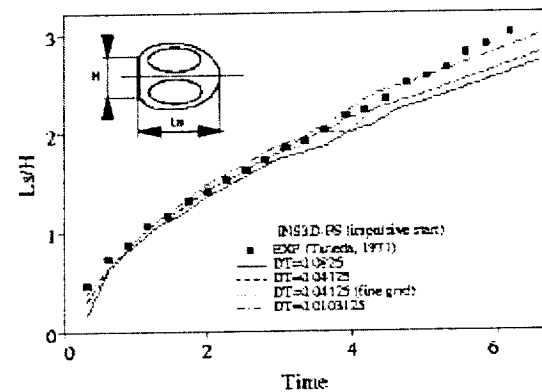
These unsteady computations using a time step size of 0.0825 was completed in two hours of CPU time on single processor Cray-J90. Computational results using the artificial compressibility code, INS3D-UP, are presented in figure 7. Results using two different artificial compressibility parameters, BETA, are compared. For time accurate solutions, sub-iterations should be terminated when the divergence of the velocity reached a specified error limit. In reality this will impose heavy burden on available computing resources. Therefore, the maximum number of sub-iterations, NSUB, is artificially fixed at 10 and 40 for the present experiment. With 10 subiterations, the



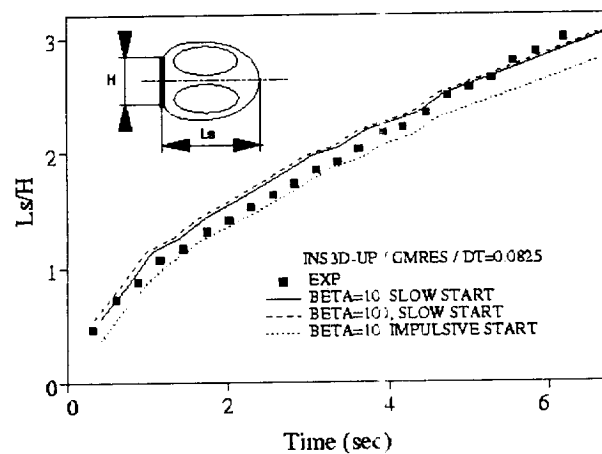
incompressibility conditions is not fully satisfied at each physical time step resulting in large error as time progresses. Computing time requirement using line relaxation scheme is large ranging 4 and 10 hours of CPU time for 10 and 40 subiterations respectively on a single processor Cray-J90 computer. It is observed that for engineering applications, a fast convergence scheme is necessary at each physical time step in order to meet incompressibility condition within reasonable accuracy. Otherwise, artificial compressibility method with line relaxation scheme can be expensive for 3D time-accurate computations. In order to alleviate this difficulty, GMRES-ILU(0) solver is implemented in INS3D-UP at the expense of increasing memory requirement. The results shown in figure 8 were obtained with less than 4 hours of Cray-J90 computer. The agreement between the computed results and experimental data is better. With GMRES-ILU(0) solver, the mass flow ratio between inflow and exit is always satisfied. In addition, the discrepancies between numerical results are very small when two different values of artificial compressibility parameter were used.



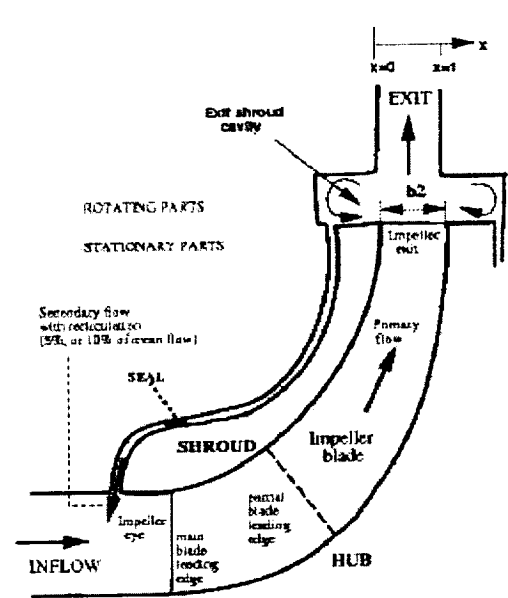
**Figure 6.** Evaluation movement of stagnation point from INS3D-UP calculation with line-relaxation scheme.



**Figure 7.** Effects of time-step size for impulsive start.



**Figure 8 .** Evaluation movement of stagnation point from INS3D-UP calculation with GMRES-ILU(0) scheme.



**Figure 9.** Schematic view of an advanced pump impeller cross-section.

When a fast converging scheme, such as a GMRES-ILU(0) solver, was implemented into artificial compressibility method, reasonable agreement was obtained between computed results and experimental data. Memory requirement of this scheme is the major drawback for three-dimensional large-scale applications. However, using parallel computing platforms, such as SGI Origin systems, memory requirement may not be a significant issue. The line-relaxation scheme in artificial compressibility method becomes very expensive for time accurate computations and could lead to erroneous solutions if incompressibility is not enforced in each time step. The pressure projection method is usually more expensive for steady state solutions due to the time required for the Poisson equation for pressure. For cases where very small physical time step is required, the pressure projection method was found to be computationally efficient since it does not require subiterations procedure. However, the governing equations are not fully coupled as in the artificial compressibility approach, and this may affect the robustness and limit the maximum allowable time step size for complicated geometries in engineering applications.

## PUMP TECHNOLOGY FOR LIQUID ROCKET ENGINE

Until recently, the high performance-pump design process was not significantly different from that of 30 years ago. During the past 30 years a vast amount of experimental and operational experience has demonstrated that there are many important features of pump flows which are not accounted for in the current semi-empirical design process. Pumps being designed today are no more technologically advanced than those designed for the Space Shuttle Main Engine (SSME). During that same time span huge strides have been made in computers, in numerical algorithms, and in physical modeling. The major accomplishment of this work is to extend the CFD technology to validate advanced CFD codes on pump flows and to demonstrate their value to the pump designer. Rocket pumps involve full and partial blades, tip leakage, and an exit boundary to diffuser. In addition to these geometric complexities, a variety of flow phenomena are encountered in turbopump flows. These include turbulent boundary layer separation, wakes, transition, tip vortices, three-dimensional effects, and Reynolds number effects. In order to increase the role of Computational Fluid Dynamics (CFD) in the design process, the CFD analysis tools must be evaluated and validated so that designers gain confidence in their use.

The incompressible Navier-Stokes flow solver, INS3D-UP, has been validated for pump component analysis. In this validation effort, computed results obtained from a rocket-pump inducer simulation were compared with experimental data. Further details can be found in the paper by Kiris et al (1993). The resulting computational procedure was applied to the flow through the SSME High Pressure Fuel Turbopump impeller and to the development of an advanced pump impeller (Kiris and Kwak 1994). The results from the advanced-pump impeller-flow analysis are presented next.

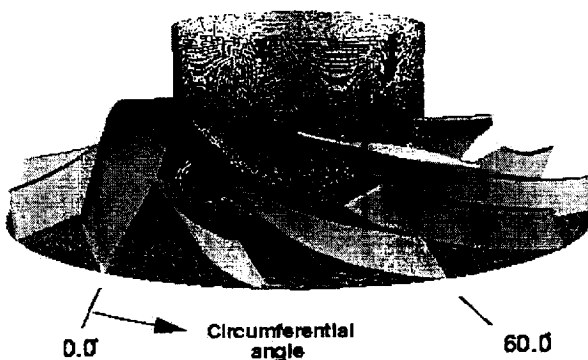


Figure 10. Advanced pump impeller computational grid on the hub surface.

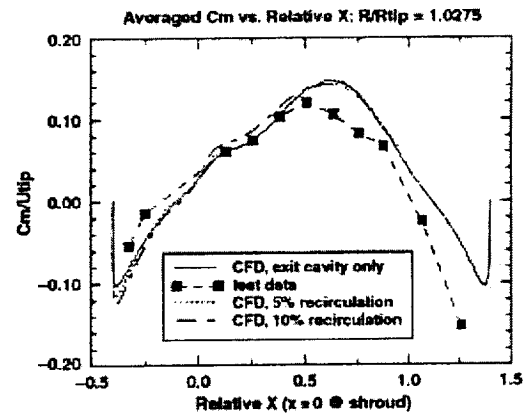
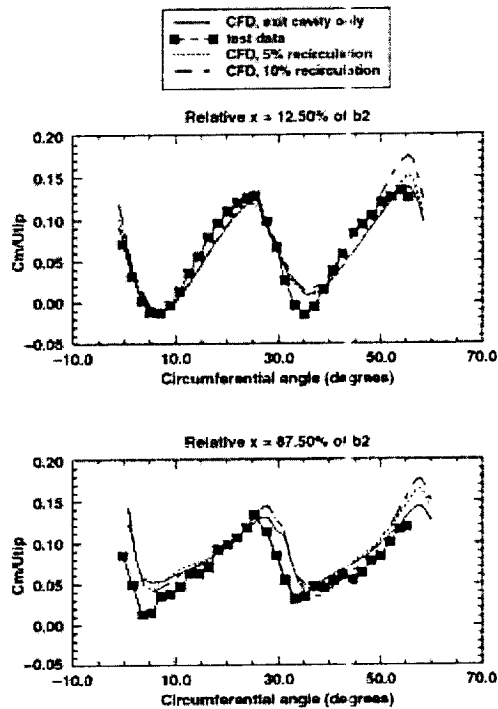


Figure 11. Comparison of circumferentially averaged meridional velocity at the impeller exit.

In Figure 9, a cross-sectional view of the advanced-pump impeller is shown schematically. The computational model of this pump includes the impeller and the exit cavity region. Figure 10 shows the computational grid near the hub

region of the impeller. The impeller design flow rate is 1,205 gal/min with a design speed of 6,322 rpm. The Reynolds Number for this calculation was 181,283 per inch. In Figure 11, the meridional velocity obtained from steady-state calculations in the rotating reference frame is shown at the impeller discharge. A relative x-distance is measured from the shroud to hub, where  $x=1.0$  is the hub. The meridional velocities,  $C_m$ , were integrated along a radial strip for each constant  $x$  position and they were non-dimensionalized by the wheel speed of 249.5 ft/sec. The meridional velocity distribution for 5% and 10% recirculation from the exit shroud cavity were also plotted. When the exit shroud cavity has leakage to the impeller eye, the velocity peak at the impeller exit moves toward to the center of the  $b_2$  width, where  $b_2$  is defined as the blade height at the impeller exit (see figure 9). However, the shroud leakage has only minor effects on the solution at impeller exit (Figure 11).

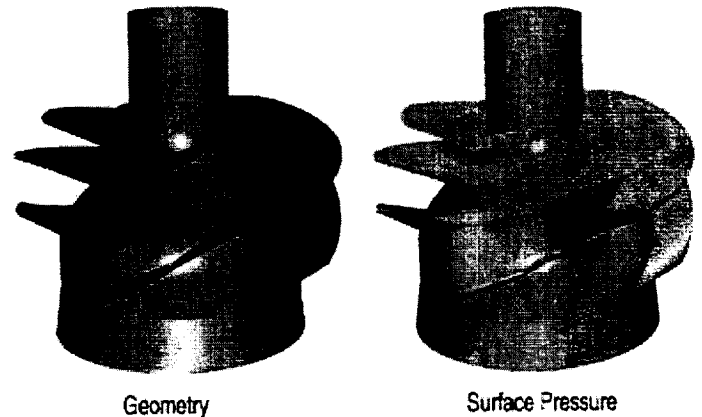


**Figure 12.** Comparison of blade-to-blade meridional velocity at the impeller exit.

In Figure 11, the symbols represent experimental data, and the lines represent  $C_m$  distributions for the flow with vaneless space at the exit of the impeller. The test data shows that the peak is closer to the center of the  $b_2$  width. The discrepancy between the computed results and experimental data is partially due to the recirculation flow in the hub cavity. The leakage at the hub cavity leads to a stronger recirculation region which shifts the velocity peak to the center of the  $b_2$  width. Since the CFD analysis did not include the leakage at the hub cavity, the predicted recirculation region in the vaneless space is not as strong as in the experimental study.

Figure 12 shows blade-to-blade velocity distributions at the impeller exit. The blade-to-blade velocity distribution illustrates the impeller-exit flow distortion. Symbols represent the experimental data and the lines represent computed results. The jet-wake like pattern, which produces an unsteady load in the diffuser vanes, was captured at both meridional locations. Overall, the numerical results compare reasonably well with the experimental data.

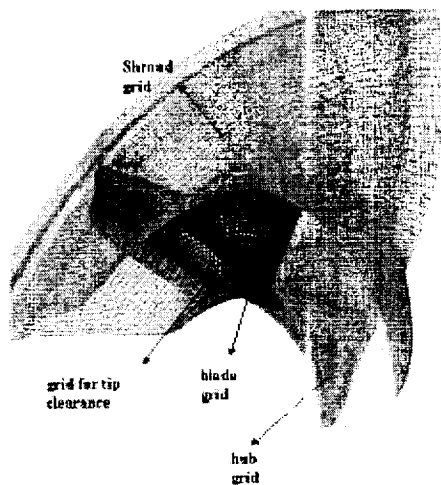
More recently, an unsteady-flow simulation capability utilizing overset grid approach for a multi-component turbopump geometry was developed at NASA-Ames Research Center. The motivation of this effort was primarily based on two elements. First, the entire turbo pump simulation is intended to provide a computational framework for the design and analysis of an entire liquid rocket engine fuel supply system. The second motivation for this research was to support the design of liquid rocket systems for development of space transportation systems. Since the



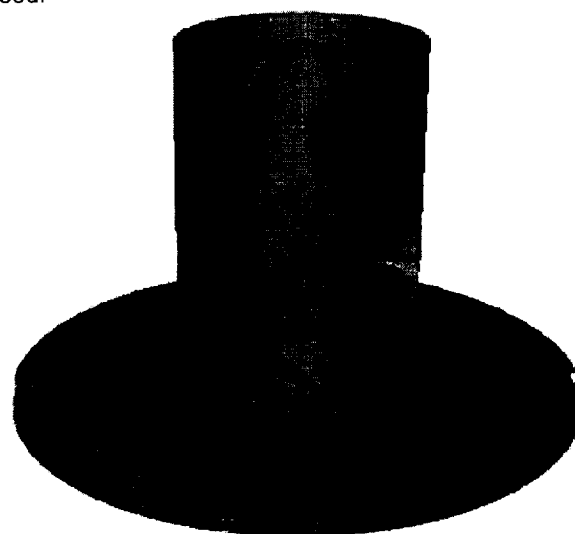
**Figure 13.** Geometry and surface pressure for a pump inducer.

space launch systems in the near future are likely to rely on liquid rocket engines, increasing the efficiency and reliability of the engine components is an important task. A substantial computational time reduction for these 3D unsteady flow simulations is required to reduce the design-cycle time of the pumps. Part of this speed up will be due to enhancements in computer hardware. The remaining portion of the speed-up must be contributed by advances in algorithms and by efficient parallel implementations. The following section outlines the initial effort and steps taken in order to reach this speed-up.

The geometry for a typical liquid oxygen pump has various rotating and stationary components, such as flow-straightener, inducer, impeller, diffuser, where the flow is extremely unsteady. Figure 13 shows the geometry and computed surface pressure of the inducer from steady-state components analysis. When rotating and stationary parts are included, time-dependent simulations need to be carried out due to relative motion of the components. To handle the geometric complexity, an overset grid approach is used.



**Figure 14.** Overset grid system for the impeller long blade section with tip clearance.



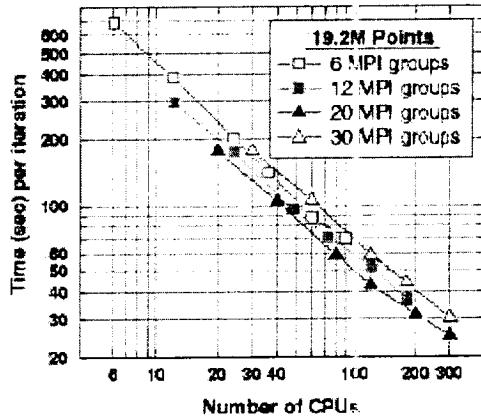
**Figure 15.** Geometry of SSME-rig1 shuttle upgrade pump impeller

The overset structured grid approach to flow simulation has been utilized to solve a variety of problems in aerospace, marine, biomedical and meteorological applications (Chan 2002). Flow regimes can range from simple steady flows as that of a commercial aircraft, to unsteady three-dimensional flows with bodies in relative motion, as in the case of turbopump configurations. A geometrically complex body is decomposed into a number of simple grid components, as shown in figure 14. In Figure 14, only long-blade impeller section is shown. For the entire configuration including inlet guide vanes, impeller blades and diffuser blades as shown in Figure 15, the computational grid has been generated by using 34.3 Million grid points with 114 zones. The freedom to allow neighboring grids to overlap arbitrarily implies that these grids can be created independently from each other and each grid is typically of high quality and nearly orthogonal. Connectivity between neighboring grids is established by interpolation at the grid outer boundaries (Meakin 2001). Addition of new components to the system and simulating arbitrary relative motion between multiple bodies are achieved by establishing new connectivity without disturbing the existing grids. Scalability on parallel compute platforms is naturally accomplished by the already decomposed grid system. For certain problems, it is more efficient to gather the grids into groups of approximately equal sizes for parallel processing.

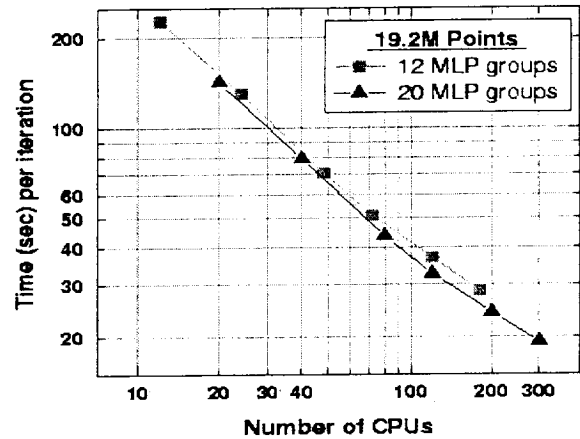
The performance of two different approaches in implementing multi-level parallelism of the INS3D code is reported in this section. The first approach is a hybrid MPI/OpenMP and the second approach is Multi Level Parallelism (MLP) developed at NASA-Ames Research Center (Taft 2000). The first approach is obtained by using message-passing interface (MPI) for inter-zone parallelism, and by using OpenMP directives for intra-zone parallelism. INS3D-MPI is based on the explicit message-passing interface across MPI groups and is designed for coarse grain parallelism. The primary strategy is to distribute the zones across a set of processors. During the iteration, all the processors would exchange boundary condition data between processors whose zones shared interfaces with zones on other processors. A simple master-worker architecture was selected because it is relatively simple to implement and it is

a common architecture for parallel CFD applications. All I/O was performed by the master MPI process and data was distributed to the workers. After the initialization phase is complete, the program begins its main iteration loop.

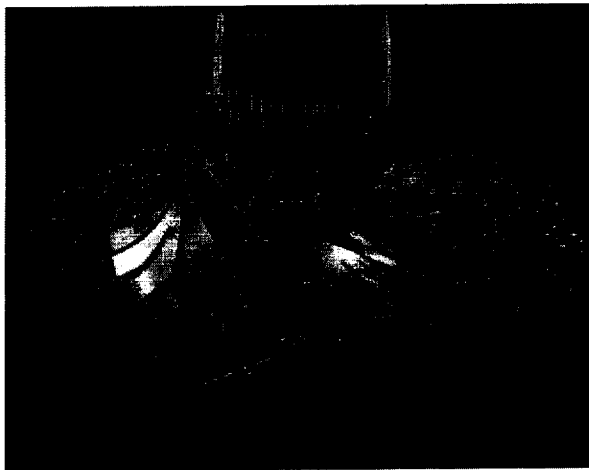
The MLP approach differs from the MPI/OpenMP approach in a fundamental way in that it does not use messaging at all. All data communication at the coarsest and finest level is accomplished via direct memory referencing instructions, however, this can only be executed on shared-memory computers. The coarsest level parallelism is implemented by spawning independent processes via the standard UNIX fork. The advantage of this approach over the MPI procedure is that the user does not have to change the initialization section of the large production code. Library of routines are used to initiate forks, to establish shared memory arenas, and to provide synchronization primitives. The boundary data for the overset-grid system is updated in the shared memory arena by each process. Other processes access the data from the arena as needed. Figure 16 and figure 17 show the speed-up for the SSME impeller computations using 19.2 million grid points by using MPI/OpenMP and MLP strategies, respectively.



**Figure 16.** Time (sec) per iteration for SSME impeller computations using INS3D-MPI/OpenMP.



**Figure 16.** Time (sec) per iteration for SSME impeller computations using INS3D-MLP.



**Figure 17.** Snapshots of particle traces and pressure surfaces from unsteady turbopump computations.

Using the MLP parallel implementation, time-accurate computations for the SSME-rig1 configuration have been carried out on SGI Origin 2000 and 3000 platforms. Instantaneous snapshots of particle traces and pressure surfaces from these computations are shown in Figure 17. The initial conditions for these simulations used flow at

rest, and then the impeller started to rotate impulsively. Three full impeller rotations were completed in the simulations using 34.3 million grid points. Using 128 SGI Origin 3000 CPUs, one impeller rotation was completed in less than 3.5 days. This capability is needed to support the design of pump sub-systems for advanced space transportation vehicles that are likely to involve liquid propulsion systems. To date, computational tools for design/analysis of turbopump flows are based on relatively lower fidelity methods. An unsteady, three-dimensional viscous flow analysis tool involving stationary and rotational components for the entire turbopump assembly has not been available for real-world engineering applications. The present effort provides developers with information such as transient flow phenomena at start up, and non-uniform inflows, and will eventually impact on system vibration and structures.

## VENTRICULAR ASSIST DEVICE

Approximately 20 million people worldwide suffer annually from congestive heart failure (CHF), a quarter of them in America alone. In the United States, an alarmingly low 2,000 to 2,500 donor hearts are available each year. One potential approach to improve this situation is to use a mechanical device to boost or to create blood flow in patients suffering from hemodynamic deterioration; that is, loss of blood pressure and lowered cardiac output. The goal of this device can be to replace the natural heart, i.e. total artificial heart, or to assist an ailing heart, i.e. ventricular assist device (VAD). In either approach, the device can be used to bridge the gap while waiting for a matching donor heart for transplantation. However, to ease the shortage of donor hearts, making these devices suitable for long-term or permanent use would be an ultimate goal.

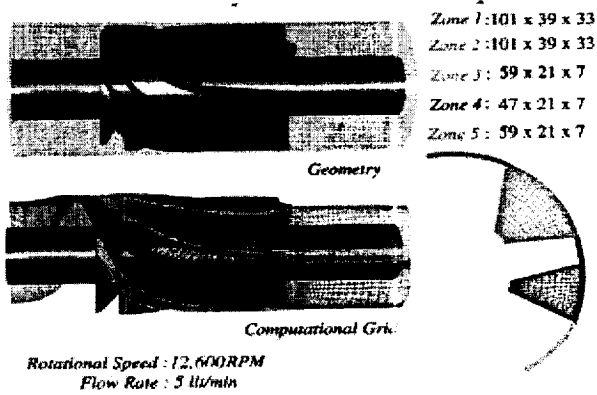
Another benefit of an assist device is the potential for providing time for the natural heart to recover. In some patients, it has been observed that the natural heart can recover by unloading the pumping requirement through the use of a VAD. In what conditions this might happen is not very well quantified at this time and should involve physiological particulars of patients among other factors. From pump technology point of view, the challenge is to design a device which can deliver the required blood circulation while not adversely impacting human physiological conditions.

Requirements for a VAD related to fluid dynamics are demanding such as: simplicity and reliability; small size for ease of implantation; pumping capacity to supply 5 liter/min of blood against 100 mmHg pressure; high pumping efficiency to minimize power requirements; and minimum hemolysis and thrombus formation. In addition to fluid dynamic issues, there are many other important aspects to be taken care of such as material compatibility with the human body, controls and implantation procedures. Due to the complexity of the flow physics and the delicate operating conditions, an empirical approach to quantify the flow phenomena in a VAD is very time consuming and expensive, especially to study many design variations. CFD simulation tools hold the potential to be invaluable for the development of these devices. In this section, the discussion is focused on how fluid dynamic issues of VAD can be resolved a computational analysis, which is extremely challenging. Flow is unsteady and involves moving parts. For a complete analysis of a VAD, a simulation of the human circulatory system has to be coupled to the device in use. However, for the purpose of developing mechanical components, a truncated circulation system can be modeled. For example, empirical inflow condition can be specified at the inlet of a VAD. Even with this type of simplifications, computational approach can produce flow field data in great detail, thus shedding lights to obtain a better understanding of the dominant flow physics produced by an artificial device. Especially, computational analysis can be utilized to optimize the design of mechanical devices at a significantly lower cost and time than required by an empirical approach.

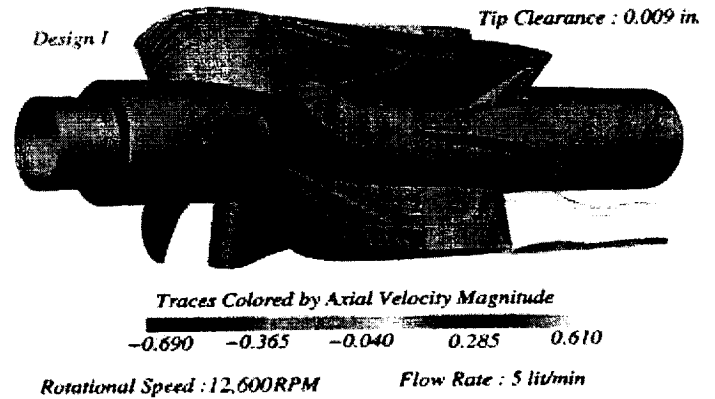
In 1989, NASA Johnson Space Center (JSC) began a joint project with the DeBakey Heart Center of the Baylor College of Medicine (BCM) in Houston to develop a new implantable prototype LVAD system. This LVAD is based on a fast rotating axial pump requiring a minimum number of moving parts. To make it implantable, the device has been made as small as possible, requiring a very high rotational speed. The computational procedure described in the pump section has been used to provide the designers with a view of the complicated fluid dynamic processes inside this device.

The flow through the baseline design of the VAD impeller was numerically simulated by using the INS3D-UP flow solver and a steady rotating frame of reference. Zonal multiblock grids were used in this component analysis. The surfaces of the computational grids for the VAD baseline impeller are shown in Figure 18. The domain is divided into five zones with dimensions of  $127 \times 39 \times 33$ ,  $127 \times 39 \times 33$ ,  $59 \times 21 \times 7$ ,  $47 \times 21 \times 5$ , and  $59 \times 21 \times 7$ , respectively. Zone 1 is the region between the suction side of the partial blade and the pressure side of the full blade; the region between the pressure side of the partial blade and the suction side of the full blade is filled by zone 2; and zones 3

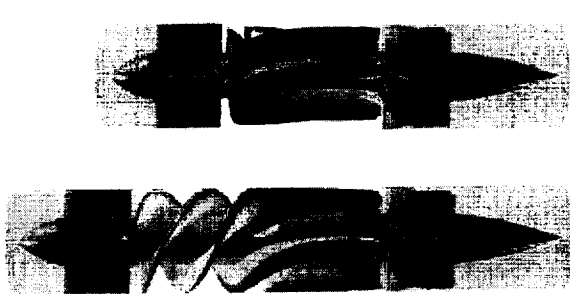
through 5 allow tip-leakage effects to be included in the computational study and occupy the regions between the impeller blade tip and the casing. At the zonal interfaces, grid points were matched one-to-one. For all zones, an H-H type grid topology was used. An H-type surface grid was generated for each surface using an elliptic grid generator. The interior region of the three-dimensional grid was filled using an algebraic-grid generator coupled with an elliptic smoother. Periodic boundary conditions were used at the end points in the rotational direction. The design flow of this impeller is 5 liters per minute and the design speed is 12,600 revolutions per minute (rpm). The problem was non-dimensionalized by the tube diameter (0.472 inches) and the impeller tip-velocity. The solution was considered converged when the maximum residual had dropped at least five orders of magnitude. Figure 19 shows the flow pattern near the suction side and pressure side of the baseline impeller blades.



**Figure 18.** Computational grid for the baseline model of DeBaKey VAD.



**Figure 19.** Flow pattern near the suction and pressure sides of VAD baseline impeller full blade.



**Figure 20.** Pressure surfaces of the baseline design (top) and new impeller design.

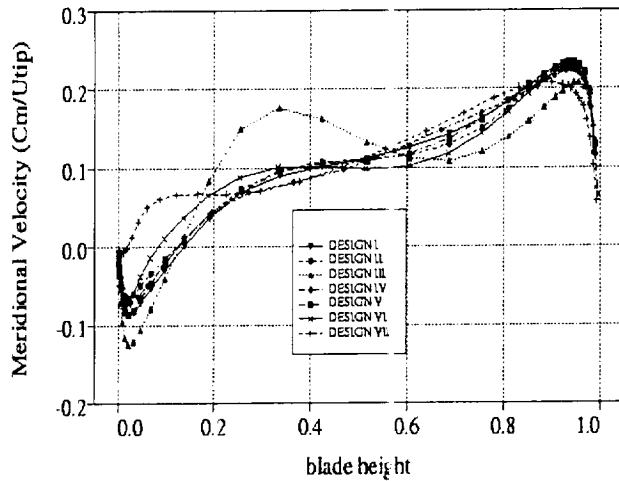


**Figure 21.** Pressure surfaces of the baseline design (top) and new impeller design.

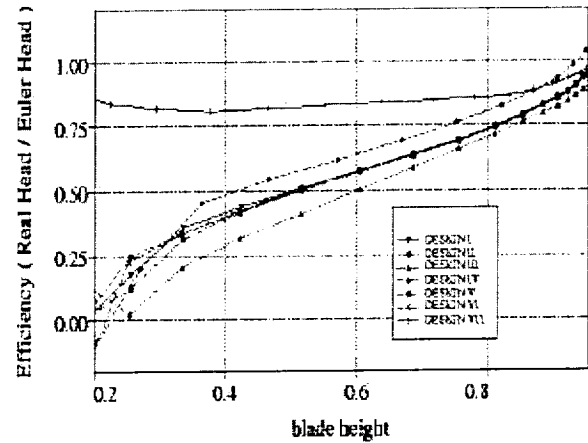
A parametric study was performed to optimize the impeller blade shape and the tip clearance. Initially, three different impeller-blade designs with a tip clearance of 0.009 inches were analyzed. Then baseline blade shape was analyzed with two tip clearances; the tip clearance of 0.0045 inches shows better hydrodynamic performance in terms of efficiency and head coefficient than with a tip clearance of 0.009 inches.

Using this design with a tip clearance of 0.0045 inches as the baseline impeller design, ideas from rocket propulsion were introduced to develop a new implantable VAD. In collaboration with Micromed Technologies and NASA-JSC engineering team and BCM researchers, a new design consisting of the baseline impeller plus an

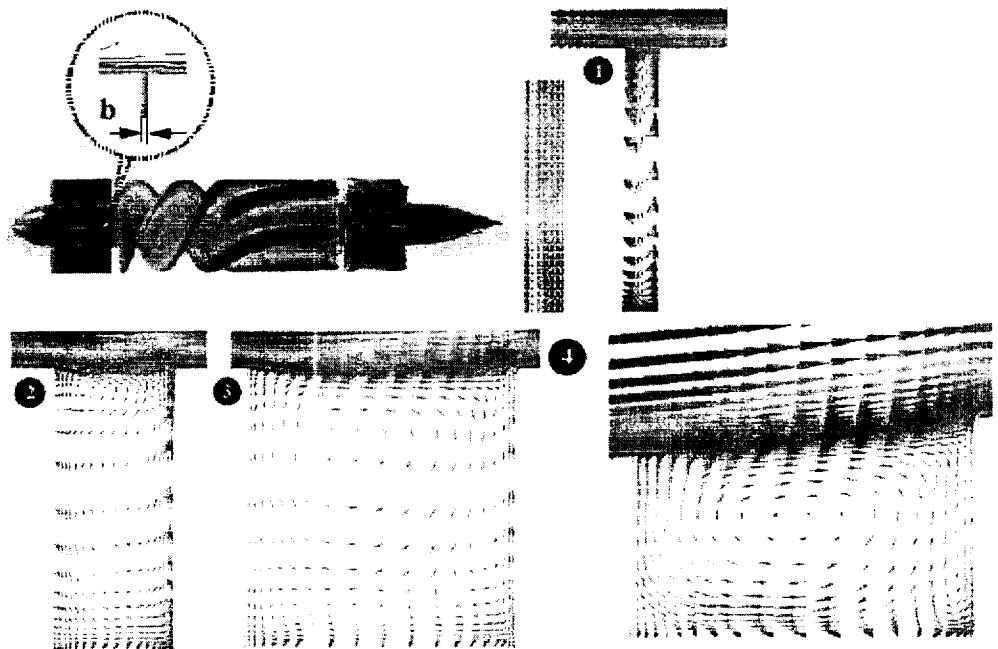
inducer was investigated. The hub and blade surfaces of the baseline impeller and the new impeller, colored by nondimensionalized pressure, are shown in Figure 20. The pressure gradient across the blades, due to the action of centrifugal force, and the pressure rise from inflow to outflow are shown. The inducer provides a sufficient pressure rise to the flow in order to prevent the cavitation on the impeller blades. Figure 21 shows the particle traces through the new impeller design. The traces are colored by the relative total-velocity magnitude. The particles were released near the inducer leading edge, the hub, the inducer blade pressure side, and the tip regions. The swirling motion of the particles indicates a secondary flow region between the partial and the full blades. The particles released near the pressure side of the blade indicate a radial velocity component inside the blade boundary layer. The particles tend to flow from the hub to the tip of the blade. The particles near the inducer leading edge and full blade trailing edge indicate the presence of back flow.



**Figure 22.** Meridional velocity distribution along impeller blade height of various designs.



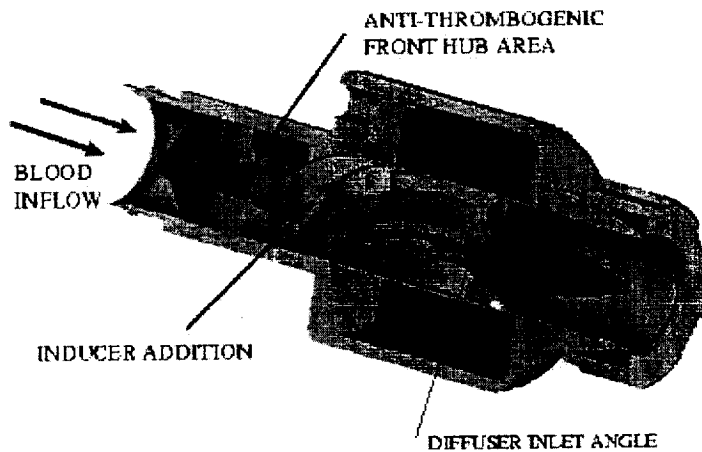
**Figure 23.** Hydrodynamic efficiency distribution along impeller blade height of various designs.



**Figure 24.** Velocity vectors inside various bearing geometries.



Parametric studies to eliminate the back flow near the hub region by tapering the hub surface have also performed for this configuration. Figure 22 shows the circumferentially averaged meridional velocity distribution along the blade height for various designs. The original blade design is referred to as Design I. Design II has less blade curvature than Design I in the trailing edge region, and Design III has more blade curvature than Design I. In Design IV, the blade shape for Design I is kept and the tip clearance is reduced from 0.009 inches to 0.004 inches. In Design V, the hub region has the blade shape for Design I and the tip region has the blade shape for Design II. In this design, the impeller blades have backward lean near the trailing edge region. In Design VI, the blades have forward lean which includes Design III in the hub region and Design I in the tip region. Design VII has small tip clearance gap, Design I blade shape and an inducer geometry upstream of impeller blades. In Figure 22, all designs except Design VII showed back flow near the hub region. The back flow has been reduced with forward blade lean which is suggested as a design change. Figure 23 shows the efficiency curves for these design variations. The inducer addition clearly shows substantial improvement in the hydrodynamic efficiency.



**Figure 25.** Contribution of CFD analysis to VAD design.

Besides improving the pumping efficiency, the design of the VAD requires good wall washing near the solid walls and reducing the stagnation regions. One of the critical regions for potential blood clotting is near the bearing area between rotating and non-rotating components. Clotting can be caused in the hub area due to either high shear or stagnation, depending on the gap and configuration of the area. Figure 24 shows velocity vectors colored by velocity magnitude for four different bearing designs. Design 1 is the original baseline design with the cavity width of  $b$ . This design showed very high shear stresses near the rotating hub face and very stagnant fluid region in the lower portion of the cavity. Increasing the cavity width to  $3.5b$  (Design 2), and to  $8b$  (Design 3 and 4) showed that the recirculation was increased in the cavity. In order to eliminate stagnant areas in the lower portion of the cavity, the hub surface was tapered. Tapering the hub surface reduced the cavity height, accelerated the flow near the hub region, and resulted in stronger recirculation in the cavity (Design 4). A modified version of Design 4 has been adopted in the current DeBakey VAD design. Figure 25 shows the areas that the VAD design is improved by using the present CFD analysis tool. This unique insight into the internal fluid structures led to an improved heart-assist device which enabled human implantation of the device. As of June, 2002, over 160 patients have successfully received this VAD. Thus, improved designs made possible because of the current work is making a far-reaching impact on human health.

## SUMMARY

In this paper, incompressible Navier-Stokes solvers designed for three-dimensional flow simulations have been discussed. The discussion has been limited to the primitive-variable formulation as it causes fewer complications in setting the boundary conditions. Numerous computed results have been presented to illustrate the numerical procedures. Even though computer speed and memory have been increased substantially in the recent past, the speed and the memory requirements of a flow solver are still major factors affecting the turnaround time. INS3D-UP,

which is an upwind finite-difference code based on an artificial compressibility approach, has been being applied to a wide variety of applications for steady-state, time-accurate and rotational-steady solutions. INS3D-FS, which is based on a pressure projection method using a finite volume discretization on staggered grids, was written solely for solving time-dependent problems. These solvers have been utilized in many applications of major engineering significance. As an example, an efficient and robust solution procedure for 3-D turbopump analyses and its spin-off application to VAD impeller has been presented. The flow through an advanced turbopump impeller and SSME rig-1 configuration have been successfully simulated. The validated solution procedure was then applied to the development of the DeBaKey VAD. Various design improvements were made through the use of this computational tool. For example, the addition of an inducer dramatically increased pumping efficiency, thereby reducing the hemolysis to an acceptable level for human use, and an optimum cavity redesign practically removed thrombus formation in the bearing area. Overall, the VAD development was expedited by extending the incompressible-flow simulation procedure originally developed for a rocket pump, thus enabling human implantation. The final measure of success has been demonstrated through successful human implantations.

## ACKNOWLEDGEMENTS

The authors would like to thank Micromed Technologies, NASA Johnson Space Center and Baylor College of Medicine VAD Team for providing the VAD geometry, and the clinical results.

## REFERENCES

- Belov, A. Martinelli, L. and Jameson, A.**, "A New Implicit Algorithm with Multigrid for Unsteady Incompressible Flow Calculations," AIAA Paper 95-0049, 1995.
- Chan, W. M., Gomez, R. J., Rogers, S. E. and Buning, P. G.**, "Best Practices in Overset Grid Generation," AIAA Paper 2002-3191, 32nd AIAA Fluid Dynamics Conference, St. Louis, Missouri, June, 2002.
- Chang, J. L. C., and Kwak, D.**, "On the Method of Pseudo Compressibility for Numerically Solving Incompressible Flows," AIAA Paper 84-0252, AIAA 22nd Aerospace Sciences Meeting, Reno, NV, January 9-12, 1984.
- Chang, J.L.C., Kwak, D., Rogers, S. E. and Yang, R-J**, "Numerical Simulation Methods of Incompressible Flows and an Application to the Space Shuttle Main Engine," Int. J. Numerical Method in Fluids, Vol. 8, pp. 1241-1268, 1988.
- Chen, Y.S., Shang, H.M and Chen, C.P.**, "Unified CFD Algorithm with a Pressure Based Method," 6th Int'l. Symposium on Comp. Fluid Dyn., Sept 4-8, 1995, Lake Tahoe, NV.
- Choi, D. and Merkle, C.L.**, "Application of Time-Iterative Schemes to Incompressible Flow," AIAA J., Vol. 23, No. 10, 1518-1524, 1985.
- Chorin, A. J.**, "A Numerical Method for Solving Incompressible Viscous Flow Problems," J. Comp. Phys., Vol. 2, pp.12-26, 1967.
- Chorin, A.J.**, "Numerical Solution of Navier-Stokes equations," Mathematics of Computation, Vol. 22, No. 104, 745-762, 1968.
- Gresho, M. P. and Sani, R. L.**, "On Pressure Boundary Conditions for the Incompressible Navier-Stokes Equations," Int. J. Numerical Methods in Fluids, Vol. 7, pp. 1111-1145, 1987.
- Hafez, M. and Oshima, K.**, ed. Computational Fluid Dynamics 1985, John Wiley and Sons, 1995.
- Harlow, F. H. and Welch, J. E.**, "Numerical Calculation of Time-Dependent Viscous Incompressible Flow with Free Surface," Phys. Fluids, Vol. 8, No. 12, pp. 2182-2189, 1965.
- Hirsch, C.**, "Numerical Computation of Internal and External Flows," John Wiley & Sons, 1988.
- Kiris, C., Chang, L., Kwak, D., and Rogers, S. E.**, "Incompressible Navier-Stokes Computations of Rotating Flows," AIAA Paper No. 93-0678, 1993.
- Kiris, C. and Kwak, D.**, "Progress in Incompressible Navier-Stokes Computations for the Analysis of Propulsion Flows," NASA CP 3282, Vol II, Advanced Earth-to-Orbit Propulsion Technology, 1994.
- Kiris, C. and Kwak, D.**, "Numerical Solution of Incompressible Navier-Stokes Equations Using a Fractional-Step Approach," AIAA Paper 96-2089, AIAA 27th Fluid Dynamics New Orleans, LA, June 17-20, 1996.
- Kiris, C., Kwak, D., and Chan, W.**, "A Three-Dimensional Parallel Time-Accurate Turbopump Simulation Procedure Using Overset Grid Systems," 2<sup>nd</sup> International Conference on Computational Fluid Dynamics, Sydney, July 15-19, 2002.
- Kwak, D., Chang, J. L. C., Shanks, S. P., and Chakravarthy, S.**, "A Three-Dimensional Incompressible Navier-Stokes Flow Solver Using Primitive Variables," AIAA J, Vol. 24, No. 3, pp 390-396, Mar. 1986.

- Kwak, D.**, "Computation of Viscous Incompressible Flows," von Karman Institute for Fluid Dynamics, Lecture Series 1989-04. Also NASA TM 101090, March 1989.
- MacCormack, R. W.**, "Current Status of Numerical Solutions of the Navier-Stokes Equations," AIAA Paper 85-0032, 1985.
- Meakin, R. L.**, "Object X-rays for Cutting Holes in Composite Overset Structured Grids," AIAA Paper 2001-2537, 15th AIAA Computational Fluid Dynamics Conference, Anaheim, California, June, 2001.
- Merkle, C. L. and Athavale, M.**, "Time-Accurate Unsteady Incompressible Flow Algorithms Based on Artificial Compressibility," AIAA Paper 87-1137, 1987.
- Patankar, S.V.**, Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Co., New York, 1980.
- Peyret, R. and Taylor, T.**, Computational Methods for Fluids, Springer-Verlag, New York, 1983.
- Rizzi, A. and Eriksson, L.-E.**, "Computation of Inviscid Incompressible Flow with Rotation," J. Comp. Phys., Vol. 153, pp 275-312, 1985.
- Rogers, S. E. and Kwak, D.**, "An Upwind Differencing Scheme for the Time-Accurate Incompressible Navier-Stokes Equations," AIAA Paper 88-2583, AIAA 6th Applied Aerodynamics Conference, Williamsburg, VA, June 6-8, 1988.
- Rogers, S. E., Kwak, D. and Kiris, C.**, "Steady and Unsteady Solutions of the Incompressible Navier-Stokes Equations," AIAA J. Vol. 29, No. 4, 603-610, April 1991.
- Rogers, S. E.**, "A Comparison of Implicit Schemes for the Incompressible Navier-Stokes Equations and Artificial Compressibility," AIAA J., Vol. 33, No. 10, Oct. 1995.
- Rosenfeld, M., Kwak, D. and Vinokur, M.**, "A Fractional Step Solution Method for the Unsteady Incompressible Navier-Stokes Equations in Generalized Coordinate Systems," J. Comp. Phys., Vol. 94, No.1, pp 102-137, May, 1991.
- Steger, J. L. and Kutler, P.**, "Implicit Finite-Difference Procedures for the Computation of Vortex Wakes," AIAA J., Vol. 15, No. 4, pp. 581-590 Apr. 1977.
- Taft, J. R.**, "Performance of the OVERFLOW-MLP and LAURA-MLP CFD Codes and the NASA Ames 512 CPU Origin Systems," HPCC/CAS 2000 Workshop, NASA Ames Research Center, 2000.
- Taneda, S., and Honji, H.**, "Unsteady Flow Past a Flat Plate Normal to the Direction of Motion," J. Phys. Soc. Japan Vol. 30, pp. 262-273, 1971.
- Temam, R.**, Navier Stokes Equations, Revised Edition., North Holland, 1979.
- Yoshida, Y., and Nomura, T.**, "A Transient Solution Method for the Finite Element Incompressible Navier-Stokes Equations," Int. J. Num. Methods in Fluids, Vol. 5, pp. 873-890, 1985.